

EngiFarm

Generated by Doxygen 1.8.16



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 Class Documentation</b>	<b>5</b>
3.1 Barn Class Reference	5
3.1.1 *	5
3.1.2 *	5
3.1.3 Detailed Description	5
3.1.4 Member Function Documentation	5
3.1.4.1 getCategory()	5
3.2 BeefChickenOmelette Class Reference	6
3.2.1 *	6
3.2.2 *	6
3.2.3 Detailed Description	6
3.2.4 Constructor & Destructor Documentation	6
3.2.4.1 BeefChickenOmelette()	6
3.2.4.2 ~BeefChickenOmelette()	6
3.2.5 Member Function Documentation	7
3.2.5.1 getCategory()	7
3.2.5.2 getPrice()	7
3.2.5.3 getRecipe()	7
3.3 BeefMuttonSate Class Reference	7
3.3.1 *	7
3.3.2 *	8
3.3.3 Detailed Description	8
3.3.4 Constructor & Destructor Documentation	8
3.3.4.1 BeefMuttonSate()	8
3.3.4.2 ~BeefMuttonSate()	8
3.3.5 Member Function Documentation	8
3.3.5.1 getCategory()	8
3.3.5.2 getPrice()	8
3.3.5.3 getRecipe()	8
3.4 Cell Class Reference	9
3.4.1 *	9
3.4.2 *	9
3.4.3 *	9
3.4.4 Detailed Description	10
3.4.5 Member Enumeration Documentation	10
3.4.5.1 Category	10
3.4.6 Constructor & Destructor Documentation	10

3.4.6.1 ~Cell()	10
3.4.7 Member Function Documentation	10
3.4.7.1 getCategory()	10
3.4.7.2 getIsOccupied()	10
3.4.7.3 growGrass()	10
3.4.7.4 isFacility()	11
3.4.7.5 isGrassExist()	11
3.4.7.6 removeGrass()	11
3.4.7.7 setIsOccupied()	11
3.4.8 Member Data Documentation	11
3.4.8.1 isOccupied	11
3.5 Chicken Class Reference	12
3.5.1 *	12
3.5.2 *	12
3.5.3 Detailed Description	12
3.5.4 Constructor & Destructor Documentation	12
3.5.4.1 Chicken()	12
3.5.5 Member Function Documentation	12
3.5.5.1 makeNoise()	13
3.5.5.2 produceProduct()	13
3.5.5.3 render()	13
3.6 ChickenEgg Class Reference	13
3.6.1 *	13
3.6.2 *	14
3.6.3 Detailed Description	14
3.6.4 Member Function Documentation	14
3.6.4.1 getCategory()	14
3.6.4.2 getPrice()	14
3.7 Coop Class Reference	14
3.7.1 *	14
3.7.2 *	15
3.7.3 Detailed Description	15
3.7.4 Member Function Documentation	15
3.7.4.1 getCategory()	15
3.8 Cow Class Reference	15
3.8.1 *	15
3.8.2 *	16
3.8.3 Detailed Description	16
3.8.4 Constructor & Destructor Documentation	16
3.8.4.1 Cow()	16
3.8.5 Member Function Documentation	16
3.8.5.1 makeNoise()	16

3.8.5.2 produceProduct()	16
3.8.5.3 render()	16
3.9 CowMeat Class Reference	17
3.9.1 *	17
3.9.2 *	17
3.9.3 Detailed Description	17
3.9.4 Member Function Documentation	17
3.9.4.1 getCategory()	17
3.9.4.2 getPrice()	17
3.10 Duck Class Reference	18
3.10.1 *	18
3.10.2 *	18
3.10.3 Detailed Description	18
3.10.4 Constructor & Destructor Documentation	18
3.10.4.1 Duck()	18
3.10.5 Member Function Documentation	18
3.10.5.1 makeNoise()	19
3.10.5.2 produceProduct()	19
3.10.5.3 render()	19
3.11 DuckMeat Class Reference	19
3.11.1 *	19
3.11.2 *	20
3.11.3 Detailed Description	20
3.11.4 Member Function Documentation	20
3.11.4.1 getCategory()	20
3.11.4.2 getPrice()	20
3.12 EggProducer Class Reference	20
3.12.1 *	20
3.12.2 *	21
3.12.3 *	21
3.12.4 Detailed Description	21
3.12.5 Constructor & Destructor Documentation	21
3.12.5.1 EggProducer()	21
3.12.5.2 ~EggProducer()	21
3.12.6 Member Function Documentation	21
3.12.6.1 getKillable()	21
3.12.6.2 getProduce()	22
3.12.7 Member Data Documentation	22
3.12.7.1 canProduce	22
3.13 Facility Class Reference	22
3.13.1 *	22
3.13.2 *	23

3.13.3 Detailed Description	23
3.13.4 Constructor & Destructor Documentation	23
3.13.4.1 Facility()	23
3.13.4.2 ~Facility()	23
3.13.5 Member Function Documentation	23
3.13.5.1 isFacility()	23
3.13.5.2 isGrassExist()	23
3.14 FarmAnimal Class Reference	24
3.14.1 *	24
3.14.2 *	24
3.14.3 *	24
3.14.4 *	24
3.14.5 Detailed Description	25
3.14.6 Member Enumeration Documentation	25
3.14.6.1 Action	25
3.14.7 Constructor & Destructor Documentation	25
3.14.7.1 FarmAnimal()	25
3.14.7.2 ~FarmAnimal()	25
3.14.8 Member Function Documentation	25
3.14.8.1 eat()	25
3.14.8.2 getKillable()	25
3.14.8.3 getProduce()	26
3.14.8.4 isDead()	26
3.14.8.5 isHungry()	26
3.14.8.6 makeNoise()	26
3.14.8.7 produceProduct()	26
3.14.8.8 tick()	26
3.14.9 Member Data Documentation	26
3.14.9.1 maxTimeToGetHungry	26
3.14.9.2 timeToGetHungry	26
3.15 FarmProduct Class Reference	27
3.15.1 *	27
3.15.2 Detailed Description	27
3.16 GrassLand Class Reference	27
3.16.1 *	27
3.16.2 *	28
3.16.3 Detailed Description	28
3.16.4 Member Function Documentation	28
3.16.4.1 getCategory()	28
3.17 Horse Class Reference	28
3.17.1 *	28
3.17.2 *	29

3.17.3 Detailed Description	29
3.17.4 Constructor & Destructor Documentation	29
3.17.4.1 Horse()	29
3.17.5 Member Function Documentation	29
3.17.5.1 makeNoise()	29
3.17.5.2 produceProduct()	29
3.17.5.3 render()	29
3.18 HorseMilk Class Reference	30
3.18.1 *	30
3.18.2 *	30
3.18.3 Detailed Description	30
3.18.4 Member Function Documentation	30
3.18.4.1 getCategory()	30
3.18.4.2 getPrice()	30
3.19 Land Class Reference	31
3.19.1 *	31
3.19.2 *	31
3.19.3 Detailed Description	31
3.19.4 Constructor & Destructor Documentation	31
3.19.4.1 ~Land()	31
3.19.5 Member Function Documentation	31
3.19.5.1 growGrass()	31
3.19.5.2 isFacility()	32
3.19.5.3 isGrassExist()	32
3.19.5.4 removeGrass()	32
3.20 LinkedList< T > Class Template Reference	32
3.20.1 *	32
3.20.2 Detailed Description	33
3.20.3 Constructor & Destructor Documentation	33
3.20.3.1 LinkedList() [1/3]	33
3.20.3.2 LinkedList() [2/3]	33
3.20.3.3 LinkedList() [3/3]	33
3.20.3.4 ~LinkedList()	33
3.20.4 Member Function Documentation	33
3.20.4.1 add()	33
3.20.4.2 find()	34
3.20.4.3 findPointer()	34
3.20.4.4 get()	34
3.20.4.5 isEmpty()	34
3.20.4.6 len()	34
3.20.4.7 operator=()	34
3.20.4.8 operator[]()	34

3.20.4.9 print()	35
3.20.4.10 remove()	35
3.20.4.11 removeldx()	35
3.21 LinkedListNode< T > Class Template Reference	35
3.21.1 *	35
3.21.2 *	35
3.21.3 Detailed Description	35
3.21.4 Constructor & Destructor Documentation	36
3.21.4.1 LinkedListNode()	36
3.21.4.2 ~LinkedListNode()	36
3.21.5 Member Data Documentation	36
3.21.5.1 LinkedList< T >	36
3.22 LivingThing Class Reference	36
3.22.1 *	36
3.22.2 *	37
3.22.3 Detailed Description	37
3.22.4 Constructor & Destructor Documentation	37
3.22.4.1 LivingThing()	37
3.22.4.2 ~LivingThing()	37
3.22.5 Member Function Documentation	37
3.22.5.1 getPosition()	37
3.22.5.2 move()	37
3.22.5.3 render()	37
3.22.6 Member Data Documentation	38
3.22.6.1 nColumnCell	38
3.22.6.2 nRowCell	38
3.22.6.3 worldMap	38
3.23 MeatProducer Class Reference	38
3.23.1 *	38
3.23.2 *	39
3.23.3 Detailed Description	39
3.23.4 Constructor & Destructor Documentation	39
3.23.4.1 MeatProducer()	39
3.23.4.2 ~MeatProducer()	39
3.23.5 Member Function Documentation	39
3.23.5.1 getKillable()	39
3.23.5.2 getProduce()	39
3.24 MilkProducer Class Reference	40
3.24.1 *	40
3.24.2 *	40
3.24.3 *	40
3.24.4 Detailed Description	40



3.24.5 Constructor & Destructor Documentation	40
3.24.5.1 MilkProducer()	40
3.24.5.2 ~MilkProducer()	41
3.24.6 Member Function Documentation	41
3.24.6.1 getKillable()	41
3.24.6.2 getProduce()	41
3.24.7 Member Data Documentation	41
3.24.7.1 canProduce	41
3.25 Mixer Class Reference	42
3.25.1 *	42
3.25.2 *	42
3.25.3 Detailed Description	42
3.25.4 Member Function Documentation	42
3.25.4.1 getCategory()	42
3.26 Ostrich Class Reference	43
3.26.1 *	43
3.26.2 *	43
3.26.3 Detailed Description	43
3.26.4 Constructor & Destructor Documentation	43
3.26.4.1 Ostrich()	43
3.26.5 Member Function Documentation	43
3.26.5.1 makeNoise()	44
3.26.5.2 produceProduct()	44
3.26.5.3 render()	44
3.27 OstrichEgg Class Reference	44
3.27.1 *	44
3.27.2 *	45
3.27.3 Detailed Description	45
3.27.4 Member Function Documentation	45
3.27.4.1 getCategory()	45
3.27.4.2 getPrice()	45
3.28 Player Class Reference	45
3.28.1 *	45
3.28.2 *	46
3.28.3 Detailed Description	46
3.28.4 Constructor & Destructor Documentation	46
3.28.4.1 Player()	46
3.28.4.2 ~Player()	46
3.28.5 Member Function Documentation	46
3.28.5.1 getInventory()	46
3.28.5.2 getMoney()	46
3.28.5.3 getrecipeBook()	46

3.28.5.4 getWater()	46
3.28.5.5 grow()	47
3.28.5.6 interact()	47
3.28.5.7 kill()	47
3.28.5.8 mix()	47
3.28.5.9 render()	47
3.28.5.10 sellAll()	47
3.28.5.11 takeWater()	47
3.28.5.12 talk()	47
3.29 Point Struct Reference	48
3.29.1 *	48
3.29.2 Detailed Description	48
3.29.3 Member Data Documentation	48
3.29.3.1 x	48
3.29.3.2 y	48
3.30 Product Class Reference	49
3.30.1 *	49
3.30.2 *	49
3.30.3 Detailed Description	49
3.30.4 Member Enumeration Documentation	49
3.30.4.1 Category	50
3.30.5 Member Function Documentation	50
3.30.5.1 getCategory()	50
3.30.5.2 getPrice()	50
3.30.5.3 operator!=(())	50
3.30.5.4 operator==(())	50
3.31 Sheep Class Reference	51
3.31.1 *	51
3.31.2 *	51
3.31.3 Detailed Description	51
3.31.4 Constructor & Destructor Documentation	51
3.31.4.1 Sheep()	51
3.31.5 Member Function Documentation	51
3.31.5.1 makeNoise()	52
3.31.5.2 produceProduct()	52
3.31.5.3 render()	52
3.32 SheepMeat Class Reference	52
3.32.1 *	52
3.32.2 *	53
3.32.3 Detailed Description	53
3.32.4 Member Function Documentation	53
3.32.4.1 getCategory()	53

3.32.4.2 getPrice()	53
3.33 SideProduct Class Reference	53
3.33.1 *	53
3.33.2 *	54
3.33.3 Detailed Description	54
3.33.4 Constructor & Destructor Documentation	54
3.33.4.1 ~SideProduct()	54
3.33.5 Member Function Documentation	54
3.33.5.1 getRecipe()	54
3.34 SuperSecretSpecialProduct Class Reference	54
3.34.1 *	54
3.34.2 *	55
3.34.3 Detailed Description	55
3.34.4 Constructor & Destructor Documentation	55
3.34.4.1 SuperSecretSpecialProduct()	55
3.34.5 Member Function Documentation	55
3.34.5.1 getCategory()	55
3.34.5.2 getPrice()	55
3.34.5.3 getRecipe()	55
3.35 Truck Class Reference	56
3.35.1 *	56
3.35.2 *	56
3.35.3 Detailed Description	56
3.35.4 Member Function Documentation	56
3.35.4.1 getCategory()	56
3.36 Well Class Reference	57
3.36.1 *	57
3.36.2 *	57
3.36.3 Detailed Description	57
3.36.4 Member Function Documentation	57
3.36.4.1 getCategory()	57
3.37 World Class Reference	58
3.37.1 *	58
3.37.2 Detailed Description	58
3.37.3 Constructor & Destructor Documentation	58
3.37.3.1 World()	58
3.37.3.2 ~World()	58
3.37.4 Member Function Documentation	58
3.37.4.1 Draw()	58
3.37.4.2 Input()	58
3.37.4.3 Update()	58



# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Cell . . . . .	9
Facility . . . . .	22
Mixer . . . . .	42
Truck . . . . .	56
Well . . . . .	57
Land . . . . .	31
Barn . . . . .	5
Coop . . . . .	14
GrassLand . . . . .	27
LinkedList< T > . . . . .	32
LinkedList< FarmAnimal * > . . . . .	32
LinkedList< Product * > . . . . .	32
LinkedList< SideProduct * > . . . . .	32
LinkedList< std::string > . . . . .	32
LinkedListNode< T > . . . . .	35
LinkedListNode< FarmAnimal * > . . . . .	35
LinkedListNode< Product * > . . . . .	35
LinkedListNode< SideProduct * > . . . . .	35
LinkedListNode< std::string > . . . . .	35
LivingThing . . . . .	36
FarmAnimal . . . . .	24
EggProducer . . . . .	20
Chicken . . . . .	12
Ostrich . . . . .	43
MeatProducer . . . . .	38
Cow . . . . .	15
Duck . . . . .	18
Sheep . . . . .	51
MilkProducer . . . . .	40
Horse . . . . .	28
Player . . . . .	45
Point . . . . .	48
Product . . . . .	49
FarmProduct . . . . .	27

ChickenEgg . . . . .	13
CowMeat . . . . .	17
DuckMeat . . . . .	19
HorseMilk . . . . .	30
OstrichEgg . . . . .	44
SheepMeat . . . . .	52
SideProduct . . . . .	53
BeefChickenOmelette . . . . .	6
BeefMuttonSate . . . . .	7
SuperSecretSpecialProduct . . . . .	54
World . . . . .	58

## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Barn	5
BeefChickenOmelette	6
BeefMuttonSate	7
Cell	9
Chicken	12
ChickenEgg	13
Coop	14
Cow	15
CowMeat	17
Duck	18
DuckMeat	19
EggProducer	20
Facility	22
FarmAnimal	24
FarmProduct	27
GrassLand	27
Horse	28
HorseMilk	30
Land	31
LinkedList< T >	32
LinkedListNode< T >	35
LivingThing	36
MeatProducer	38
MilkProducer	40
Mixer	42
Ostrich	43
OstrichEgg	44
Player	45
Point	48
Product	49
Sheep	51
SheepMeat	52
SideProduct	53
SuperSecretSpecialProduct	54
Truck	56
Well	57
World	58





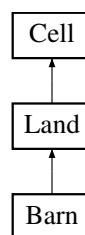
## Chapter 3

# Class Documentation

### 3.1 Barn Class Reference

```
#include <Barn.h>
```

Inheritance diagram for Barn:



#### 3.1.1 \*

Public Member Functions

- [Category getCategory \(\)](#) const

#### 3.1.2 \*

Additional Inherited Members

#### 3.1.3 Detailed Description

[Barn](#) merupakan kelas turunan dari [Land](#) yang hanya bisa ditempati oleh [Player](#) dan [MeatProducer](#)

#### 3.1.4 Member Function Documentation

##### 3.1.4.1 getCategory()

```
Category Barn::getCategory ( ) const [virtual]
```

Return kategori dari objek ini

Implements [Cell](#).

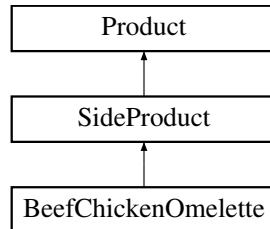
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Barn.h

## 3.2 BeefChickenOmelette Class Reference

```
#include <BeefChickenOmelette.h>
```

Inheritance diagram for BeefChickenOmelette:



### 3.2.1 \*

Public Member Functions

- [BeefChickenOmelette \(\)](#)
- [~BeefChickenOmelette \(\)](#)
- [int getPrice \(\) const](#)
- [Category getCategory \(\) const](#)
- [LinkedList< Product \\* > & getRecipe \(\)](#)

### 3.2.2 \*

Additional Inherited Members

### 3.2.3 Detailed Description

[BeefChickenOmelette](#) adalah kelas turunan dari [SideProduct](#) yang dihasilkan dengan mix [CowMeat](#) dan [ChickenEgg](#)

### 3.2.4 Constructor & Destructor Documentation

#### 3.2.4.1 BeefChickenOmelette()

```
BeefChickenOmelette::BeefChickenOmelette ( )
```

Constructor untuk inisialisasi recipe

#### 3.2.4.2 ~BeefChickenOmelette()

```
BeefChickenOmelette::~~BeefChickenOmelette ( )
```

Dtor

### 3.2.5 Member Function Documentation

#### 3.2.5.1 getCategory()

```
Category BeefChickenOmelette::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

#### 3.2.5.2 getPrice()

```
int BeefChickenOmelette::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

#### 3.2.5.3 getRecipe()

```
LinkedList<Product*>& BeefChickenOmelette::getRecipe ( ) [virtual]
```

Mengembalikan resep dari produk

Implements [SideProduct](#).

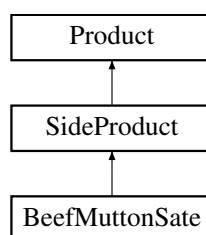
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/BeefChickenOmelette.h

## 3.3 BeefMuttonSate Class Reference

```
#include <BeefMuttonSate.h>
```

Inheritance diagram for BeefMuttonSate:



### 3.3.1 \*

Public Member Functions

- [BeefMuttonSate](#) ()
- [~BeefMuttonSate](#) ()
- int [getPrice](#) () const
- [Category](#) [getCategory](#) () const
- [LinkedList](#)< [Product](#) \* > & [getRecipe](#) ()

### 3.3.2 \*

Additional Inherited Members

### 3.3.3 Detailed Description

[BeefMuttonSate](#) adalah kelas turunan dari [SideProduct](#) yang dihasilkan dengan mix [CowMeat](#) dan [SheepMeat](#)

### 3.3.4 Constructor & Destructor Documentation

#### 3.3.4.1 BeefMuttonSate()

```
BeefMuttonSate::BeefMuttonSate ( )
```

Constructor untuk inisialisasi recipe

#### 3.3.4.2 ~BeefMuttonSate()

```
BeefMuttonSate::~~BeefMuttonSate ( )
```

Dtor

### 3.3.5 Member Function Documentation

#### 3.3.5.1 getCategory()

```
Category BeefMuttonSate::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

#### 3.3.5.2 getPrice()

```
int BeefMuttonSate::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

#### 3.3.5.3 getRecipe()

```
LinkedList<Product*>& BeefMuttonSate::getRecipe ( ) [virtual]
```

Mengembalikan resep dari produk

Implements [SideProduct](#).

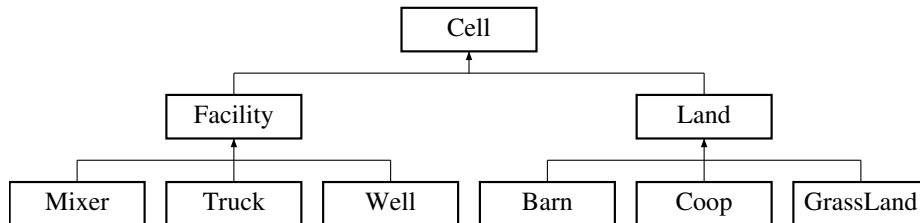
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/BeefMuttonSate.h

## 3.4 Cell Class Reference

```
#include <Cell.h>
```

Inheritance diagram for Cell:



### 3.4.1 \*

Public Types

- enum [Category](#) {  
**WELL, MIXER, TRUCK, COOP,**  
**GRASSLAND, BARN** }

### 3.4.2 \*

Public Member Functions

- virtual [~Cell](#) ()=0
- virtual bool [isFacility](#) () const =0
- virtual [Category getCategory](#) () const =0
- bool [getIsOccupied](#) ()
- void [setIsOccupied](#) (bool)
- virtual void [growGrass](#) ()
- virtual void [removeGrass](#) ()
- virtual bool [isGrassExist](#) () const =0

### 3.4.3 \*

Protected Attributes

- bool [isOccupied](#) {false}

### 3.4.4 Detailed Description

[Cell](#) adalah kelas abstrak yang merepresentasikan petak pada Engi's farm

### 3.4.5 Member Enumeration Documentation

#### 3.4.5.1 Category

```
enum Cell::Category
```

Jenis kategori [Cell](#)

### 3.4.6 Constructor & Destructor Documentation

#### 3.4.6.1 ~Cell()

```
virtual Cell::~Cell ( ) [pure virtual]
```

dtor untuk [Cell](#)

### 3.4.7 Member Function Documentation

#### 3.4.7.1 getCategory()

```
virtual Category Cell::getCategory ( ) const [pure virtual]
```

Return kategori dari objek kategori

Implemented in [Truck](#), [Barn](#), [Coop](#), [GrassLand](#), [Mixer](#), and [Well](#).

#### 3.4.7.2 getIsOccupied()

```
bool Cell::getIsOccupied ( )
```

Mengambil nilai boolean isOccupied

#### 3.4.7.3 growGrass()

```
virtual void Cell::growGrass ( ) [virtual]
```

Menambah air pada cell. Jika bertipe [Land](#) akan menumbuhkan rumput. Jika tidak, tidak akan berefek apa-apa.

Reimplemented in [Land](#).

#### 3.4.7.4 isFacility()

```
virtual bool Cell::isFacility ( ) const [pure virtual]
```

Return true jika objek adalah [Facility](#)

Implemented in [Facility](#), and [Land](#).

#### 3.4.7.5 isGrassExist()

```
virtual bool Cell::isGrassExist ( ) const [pure virtual]
```

Mengembalikan keberadaan grass jika [Cell](#) bertipe [Land](#)

Implemented in [Facility](#), and [Land](#).

#### 3.4.7.6 removeGrass()

```
virtual void Cell::removeGrass ( ) [virtual]
```

Jika [Cell](#) bertipe [Land](#) dan memiliki rumput maka rumput akan dihilangkan

Reimplemented in [Land](#).

#### 3.4.7.7 setIsOccupied()

```
void Cell::setIsOccupied (
    bool )
```

Mengganti nilai boolean isOccupied

### 3.4.8 Member Data Documentation

#### 3.4.8.1 isOccupied

```
bool Cell::isOccupied {false} [protected]
```

Flag yang menandakan cell ditempati oleh sesuatu (Player/FarmAnimal/Facility) atau tidak. True bila cell sedang ditempati oleh sesuatu.

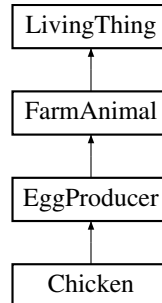
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Cell.h

### 3.5 Chicken Class Reference

```
#include <Chicken.h>
```

Inheritance diagram for Chicken:



#### 3.5.1 \*

Public Member Functions

- [Chicken](#) ([Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nCollumnCell)
- [FarmProduct](#) \* [produceProduct](#) ([Action](#))
- `std::string makeNoise () const`
- `char render ()`

#### 3.5.2 \*

Additional Inherited Members

#### 3.5.3 Detailed Description

[Chicken](#) merupakan kelas turunan dari [EggProducer](#) yang menghasilkan [ChickenEgg](#) saat diinteract

#### 3.5.4 Constructor & Destructor Documentation

##### 3.5.4.1 [Chicken\(\)](#)

```

Chicken::Chicken (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
  
```

Constructor

#### 3.5.5 Member Function Documentation



## 3.5.5.1 makeNoise()

```
std::string Chicken::makeNoise ( ) const [virtual]
```

Mengembalikan suara dari [Chicken](#)

Implements [FarmAnimal](#).

## 3.5.5.2 produceProduct()

```
FarmProduct* Chicken::produceProduct (
    Action ) [virtual]
```

Mengembalikan FarmProduk yang akan dihasilkan [Chicken](#) bila [Chicken](#) di interact

Implements [FarmAnimal](#).

## 3.5.5.3 render()

```
char Chicken::render ( ) [virtual]
```

Mengembalikan karakter yang merepresentasikan [Chicken](#) saat Hungry dan tidak Hungry

Implements [LivingThing](#).

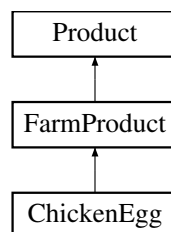
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/Chicken.h

## 3.6 ChickenEgg Class Reference

```
#include <ChickenEgg.h>
```

Inheritance diagram for ChickenEgg:



## 3.6.1 \*

Public Member Functions

- int [getPrice](#) ( ) const
- [Category getCategory](#) ( ) const

### 3.6.2 \*

Additional Inherited Members

### 3.6.3 Detailed Description

[ChickenEgg](#) adalah kelas turunan dari [FarmProduct](#) yang dihasilkan dengan interact dengan [Chicken](#)

### 3.6.4 Member Function Documentation

#### 3.6.4.1 getCategory()

```
Category ChickenEgg::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

#### 3.6.4.2 getPrice()

```
int ChickenEgg::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

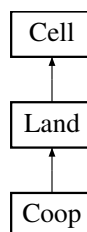
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/ChickenEgg.h

## 3.7 Coop Class Reference

```
#include <Coop.h>
```

Inheritance diagram for Coop:



### 3.7.1 \*

Public Member Functions

- [Category](#) [getCategory](#) ( ) const

## 3.7.2 \*

Additional Inherited Members

## 3.7.3 Detailed Description

[Coop](#) merupakan kelas turunan dari [Land](#) yang hanya bisa ditempati oleh [Player](#) dan [EggProducer](#)

## 3.7.4 Member Function Documentation

## 3.7.4.1 getCategory()

```
Category Coop::getCategory ( ) const [virtual]
```

Return kategori dari objek ini

Implements [Cell](#).

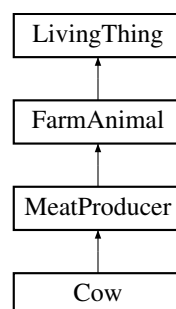
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Coop.h

## 3.8 Cow Class Reference

```
#include <Cow.h>
```

Inheritance diagram for Cow:



## 3.8.1 \*

Public Member Functions

- [Cow](#) ([Point](#) position, [Cell](#) \*\*\*&[worldMap](#), int nRowCell, int nColumnCell)
- [FarmProduct](#) \* [produceProduct](#) ([Action](#))
- std::string [makeNoise](#) () const
- char [render](#) ()

### 3.8.2 \*

Additional Inherited Members

### 3.8.3 Detailed Description

[Cow](#) merupakan kelas turunan dari [MeatProducer](#) yang menghasilkan [CowMeat](#) saat dikill

### 3.8.4 Constructor & Destructor Documentation

#### 3.8.4.1 Cow()

```
Cow::Cow (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor

### 3.8.5 Member Function Documentation

#### 3.8.5.1 makeNoise()

```
std::string Cow::makeNoise ( ) const [virtual]
```

Mengembalikan suara dari [Cow](#)

Implements [FarmAnimal](#).

#### 3.8.5.2 produceProduct()

```
FarmProduct* Cow::produceProduct (
    Action ) [virtual]
```

Mengembalikan FarmProduk yang akan dihasilkan [Cow](#) bila [Cow](#) di kill

Implements [FarmAnimal](#).

#### 3.8.5.3 render()

```
char Cow::render ( ) [virtual]
```

Mengembalikan karakter yang merepresentasikan [Cow](#) saat Hungry dan tidak Hungry

Implements [LivingThing](#).

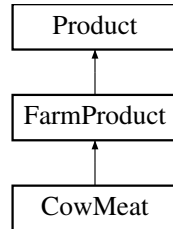
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/Cow.h

## 3.9 CowMeat Class Reference

```
#include <CowMeat.h>
```

Inheritance diagram for CowMeat:



### 3.9.1 \*

Public Member Functions

- `int getPrice () const`
- `Category getCategory () const`

### 3.9.2 \*

Additional Inherited Members

### 3.9.3 Detailed Description

`CowMeat` adalah kelas turunan dari `FarmProduct` yang dihasilkan dengan kill `Cow`

### 3.9.4 Member Function Documentation

#### 3.9.4.1 getCategory()

```
Category CowMeat::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements `Product`.

#### 3.9.4.2 getPrice()

```
int CowMeat::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements `Product`.

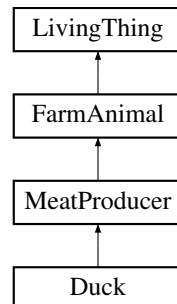
The documentation for this class was generated from the following file:

- `include/EngiFarm/Product/CowMeat.h`

### 3.10 Duck Class Reference

```
#include <Duck.h>
```

Inheritance diagram for Duck:



#### 3.10.1 \*

Public Member Functions

- [Duck](#) ([Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nCollumnCell)
- [FarmProduct](#) \* produceProduct ([Action](#))
- std::string makeNoise () const
- char render ()

#### 3.10.2 \*

Additional Inherited Members

#### 3.10.3 Detailed Description

[Duck](#) merupakan kelas turunan dari [MeatProducer](#) yang menghasilkan [DuckMeat](#) saat diinteract

#### 3.10.4 Constructor & Destructor Documentation

##### 3.10.4.1 Duck()

```
Duck::Duck (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor

#### 3.10.5 Member Function Documentation

## 3.10.5.1 makeNoise()

```
std::string Duck::makeNoise ( ) const [virtual]
```

Mengembalikan suara dari [Duck](#)

Implements [FarmAnimal](#).

## 3.10.5.2 produceProduct()

```
FarmProduct* Duck::produceProduct (
    Action ) [virtual]
```

Mengembalikan FarmProduk yang akan dihasilkan [Duck](#) bila [Duck](#) di kill

Implements [FarmAnimal](#).

## 3.10.5.3 render()

```
char Duck::render ( ) [virtual]
```

Mengembalikan karakter yang merepresentasikan [Duck](#) saat Hungry dan tidak Hungry

Implements [LivingThing](#).

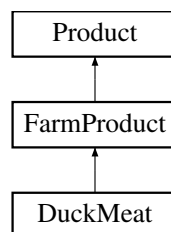
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/Duck.h

## 3.11 DuckMeat Class Reference

```
#include <DuckMeat.h>
```

Inheritance diagram for DuckMeat:



## 3.11.1 \*

Public Member Functions

- int [getPrice](#) ( ) const
- [Category getCategory](#) ( ) const

### 3.11.2 \*

Additional Inherited Members

### 3.11.3 Detailed Description

kelas turunan dari Farmproduct yang dihasilkan dengan interact dengan duck

### 3.11.4 Member Function Documentation

#### 3.11.4.1 getCategory()

```
Category DuckMeat::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

#### 3.11.4.2 getPrice()

```
int DuckMeat::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

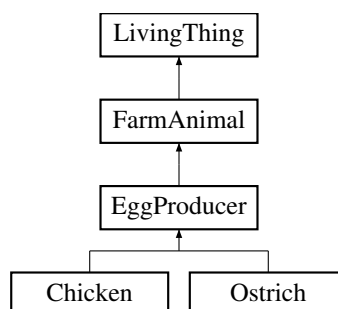
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/DuckMeat.h

## 3.12 EggProducer Class Reference

```
#include <EggProducer.h>
```

Inheritance diagram for EggProducer:



### 3.12.1 \*

Public Member Functions

- [EggProducer](#) (int \_maxTimeToGetHungry, [Point](#) position, [Cell](#) \*\*\*&[worldMap](#), int [nRowCell](#), int [nCollumnCell](#))
- virtual [~EggProducer](#) ()=0
- bool [getProduce](#) ()
- bool [getKillable](#) ()



### 3.12.2 \*

#### Protected Attributes

- bool `canProduce` {false}

### 3.12.3 \*

#### Additional Inherited Members

### 3.12.4 Detailed Description

`EggProducer` merupakan kelas abstrak turunan dari `FarmAnimal` yang tinggal di `Coop` dan menghasilkan Egg saat diinteract

### 3.12.5 Constructor & Destructor Documentation

#### 3.12.5.1 `EggProducer()`

```
EggProducer::EggProducer (
    int _maxTimeToGetHungry,
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor `maxTimeToGetHungry` dengan nilai H

#### 3.12.5.2 `~EggProducer()`

```
virtual EggProducer::~~EggProducer ( ) [pure virtual]
```

Penerusan overloading (virtual) destruktur

### 3.12.6 Member Function Documentation

#### 3.12.6.1 `getKillable()`

```
bool EggProducer::getKillable ( ) [virtual]
```

Mengembalikan false karena `EggProducer` tidak bisa di kill

Implements `FarmAnimal`.

### 3.12.6.2 getProduce()

```
bool EggProducer::getProduce ( ) [virtual]
```

Mengembalikan nilai dari canProduce

Implements [FarmAnimal](#).

## 3.12.7 Member Data Documentation

### 3.12.7.1 canProduce

```
bool EggProducer::canProduce {false} [protected]
```

Menentukan apakah [FarmAnimal](#) dapat menghasilkan produk apabila diinteract

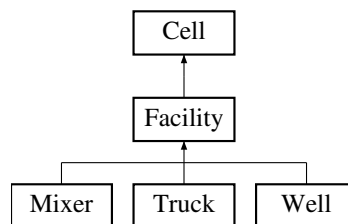
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/EggProducer.h

## 3.13 Facility Class Reference

```
#include <Facility.h>
```

Inheritance diagram for Facility:



### 3.13.1 \*

Public Member Functions

- [Facility](#) ()
- virtual [~Facility](#) ()=0
- bool [isFacility](#) () const
- bool [isGrassExist](#) () const

### 3.13.2 \*

Additional Inherited Members

### 3.13.3 Detailed Description

[Facility](#) merupakan kelas turunan dari [Cell](#) yang menampung utilitas untuk [Player](#) yaitu [Truck](#), [Mixer](#), dan [Well](#) dan tidak bisa ditempati oleh [LivingThing](#)

### 3.13.4 Constructor & Destructor Documentation

#### 3.13.4.1 Facility()

```
Facility::Facility ( )
```

Constructor untuk set isOccupied jadi true

#### 3.13.4.2 ~Facility()

```
virtual Facility::~~Facility ( ) [pure virtual]
```

Destructor [Land](#)

### 3.13.5 Member Function Documentation

#### 3.13.5.1 isFacility()

```
bool Facility::isFacility ( ) const [virtual]
```

Return true bila [Land](#) adalah sebuah facility

Implements [Cell](#).

#### 3.13.5.2 isGrassExist()

```
bool Facility::isGrassExist ( ) const [virtual]
```

Mengembalikan false

Implements [Cell](#).

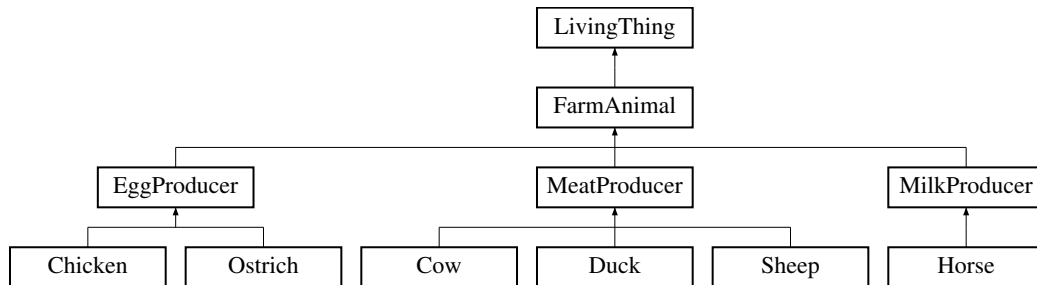
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Facility.h

### 3.14 FarmAnimal Class Reference

```
#include <FarmAnimal.h>
```

Inheritance diagram for FarmAnimal:



#### 3.14.1 \*

Public Types

- enum [Action](#) { **INTERACT**, **KILL** }

#### 3.14.2 \*

Public Member Functions

- [FarmAnimal](#) (int \_maxTimeToGetHungry, [Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nColumnCell)
- virtual [~FarmAnimal](#) ()=0
- void [tick](#) ()
- virtual [FarmProduct](#) \* [produceProduct](#) ([Action](#))=0
- virtual std::string [makeNoise](#) () const =0
- bool [isDead](#) () const
- virtual bool [getProduce](#) ()=0
- virtual bool [getKillable](#) ()=0

#### 3.14.3 \*

Protected Member Functions

- virtual void [eat](#) ()
- bool [isHungry](#) () const

#### 3.14.4 \*

Protected Attributes

- int [timeToGetHungry](#)
- const int [maxTimeToGetHungry](#)

### 3.14.5 Detailed Description

kelas [FarmAnimal](#) merupakan kelas turunan dari living thing yang dapat berupa [EggProducer](#), [MilkProducer](#), dan [MeatProducer](#)

### 3.14.6 Member Enumeration Documentation

#### 3.14.6.1 Action

```
enum FarmAnimal::Action
```

Jenis aksi yang dapat dilakukan ke [FarmAnimal](#)

### 3.14.7 Constructor & Destructor Documentation

#### 3.14.7.1 FarmAnimal()

```
FarmAnimal::FarmAnimal (  
    int _maxTimeToGetHungry,  
    Point position,  
    Cell ***& worldMap,  
    int nRowCell,  
    int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H

#### 3.14.7.2 ~FarmAnimal()

```
virtual FarmAnimal::~FarmAnimal ( ) [pure virtual]
```

Destructor [FarmAnimal](#)

### 3.14.8 Member Function Documentation

#### 3.14.8.1 eat()

```
virtual void FarmAnimal::eat ( ) [protected], [virtual]
```

Jika [FarmAnimal](#) sedang berdiri pada land dengan rumput, maka timeToDeath di set nilai semula dan timeToGdengan nilai sesuai dengan derived classnya, lalu grass di land dihapus

#### 3.14.8.2 getKillable()

```
virtual bool FarmAnimal::getKillable ( ) [pure virtual]
```

Mengembalikan true jika [FarmAnimal](#) bisa di Kill untuk menghasilkan [Product](#)

Implemented in [MilkProducer](#), [EggProducer](#), and [MeatProducer](#).

#### 3.14.8.3 `getProduce()`

```
virtual bool FarmAnimal::getProduce ( ) [pure virtual]
```

Mengembalikan true jika [FarmAnimal](#) bisa di Interact untuk menghasilkan [Product](#)

Implemented in [MeatProducer](#), [MilkProducer](#), and [EggProducer](#).

#### 3.14.8.4 `isDead()`

```
bool FarmAnimal::isDead ( ) const
```

Mengembalikan true jika `timeToDeath == 0`, lalu di destruct di main atau di class world

#### 3.14.8.5 `isHungry()`

```
bool FarmAnimal::isHungry ( ) const [protected]
```

return true apabila `timeToGetHungry <= 0`

#### 3.14.8.6 `makeNoise()`

```
virtual std::string FarmAnimal::makeNoise ( ) const [pure virtual]
```

Mengembalikan suara dari [FarmAnimal](#)

Implemented in [Chicken](#), [Cow](#), [Duck](#), [Horse](#), [Ostrich](#), and [Sheep](#).

#### 3.14.8.7 `produceProduct()`

```
virtual FarmProduct* FarmAnimal::produceProduct (
    Action ) [pure virtual]
```

Mengembalikan produk yang dihasilkan [FarmAnimal](#) apabila diinteract/dikill

Implemented in [Chicken](#), [Cow](#), [Duck](#), [Horse](#), [Ostrich](#), and [Sheep](#).

#### 3.14.8.8 `tick()`

```
void FarmAnimal::tick ( )
```

Melakukan aksi yang dilakukan [FarmAnimal](#) setiap satuan waktu

### 3.14.9 Member Data Documentation

#### 3.14.9.1 `maxTimeToGetHungry`

```
const int FarmAnimal::maxTimeToGetHungry [protected]
```

Nilai max dari `timeToGetHungry`

#### 3.14.9.2 `timeToGetHungry`

```
int FarmAnimal::timeToGetHungry [protected]
```

Waktu [FarmAnimal](#) sampai menjadi lapar

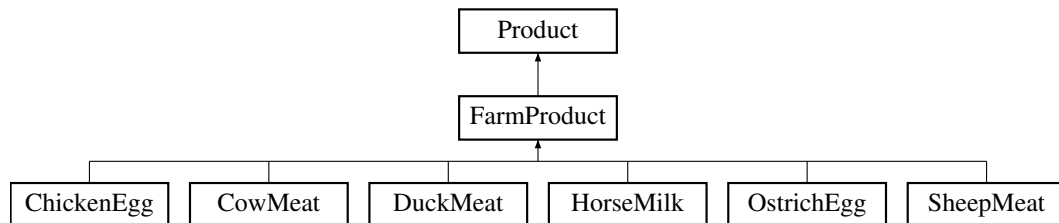
The documentation for this class was generated from the following file:

- `include/EngiFarm/FarmAnimal/FarmAnimal.h`

## 3.15 FarmProduct Class Reference

```
#include <FarmProduct.h>
```

Inheritance diagram for FarmProduct:



### 3.15.1 \*

Additional Inherited Members

### 3.15.2 Detailed Description

[Product](#) yang didapat dari hasil interact / kill

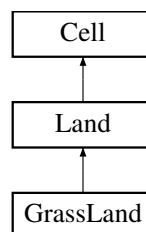
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/FarmProduct.h

## 3.16 GrassLand Class Reference

```
#include <GrassLand.h>
```

Inheritance diagram for GrassLand:



### 3.16.1 \*

Public Member Functions

- [Category getCategory \(\)](#) const

### 3.16.2 \*

Additional Inherited Members

### 3.16.3 Detailed Description

[GrassLand](#) merupakan kelas turunan dari [Land](#) yang hanya bisa ditempati oleh [Player](#) dan [MilkProducer](#)

### 3.16.4 Member Function Documentation

#### 3.16.4.1 getCategory()

```
Category GrassLand::getCategory ( ) const [virtual]
```

Return kategori dari objek ini

Implements [Cell](#).

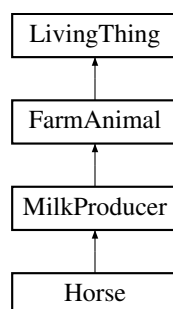
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/GrassLand.h

## 3.17 Horse Class Reference

```
#include <Horse.h>
```

Inheritance diagram for Horse:



### 3.17.1 \*

Public Member Functions

- [Horse](#) ([Point](#) position, [Cell](#) \*\*\*&[worldMap](#), int nRowCell, int nCollumnCell)
- [FarmProduct](#) \* [produceProduct](#) ([Action](#))
- std::string [makeNoise](#) () const
- char [render](#) ()



### 3.17.2 \*

Additional Inherited Members

### 3.17.3 Detailed Description

[Horse](#) merupakan kelas turunan dari [MilkProducer](#) yang menghasilkan [HorseMilk](#) saat diinteract

### 3.17.4 Constructor & Destructor Documentation

#### 3.17.4.1 Horse()

```
Horse::Horse (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor

### 3.17.5 Member Function Documentation

#### 3.17.5.1 makeNoise()

```
std::string Horse::makeNoise ( ) const [virtual]
```

Mengembalikan suara dari [Horse](#)

Implements [FarmAnimal](#).

#### 3.17.5.2 produceProduct()

```
FarmProduct* Horse::produceProduct (
    Action ) [virtual]
```

Mengembalikan FarmProduk yang akan dihasilkan [Horse](#) bila [Horse](#) di interact

Implements [FarmAnimal](#).

#### 3.17.5.3 render()

```
char Horse::render ( ) [virtual]
```

Mengembalikan karakter yang merepresentasikan [Horse](#) saat Hungry dan tidak Hungry

Implements [LivingThing](#).

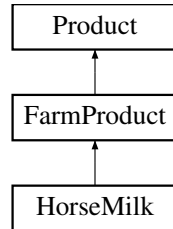
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/Horse.h

## 3.18 HorseMilk Class Reference

```
#include <HorseMilk.h>
```

Inheritance diagram for HorseMilk:



### 3.18.1 \*

Public Member Functions

- int [getPrice](#) () const
- [Category](#) [getCategory](#) () const

### 3.18.2 \*

Additional Inherited Members

### 3.18.3 Detailed Description

[HorseMilk](#) adalah kelas turunan dari [FarmProduct](#) yang dihasilkan dengan interact dengan [Horse](#)

### 3.18.4 Member Function Documentation

#### 3.18.4.1 getCategory()

```
Category HorseMilk::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

#### 3.18.4.2 getPrice()

```
int HorseMilk::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

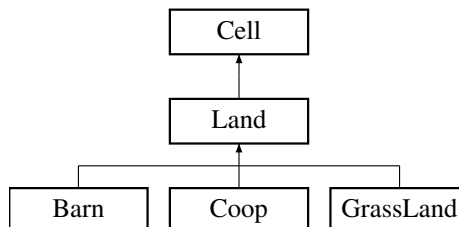
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/HorseMilk.h

## 3.19 Land Class Reference

```
#include <Land.h>
```

Inheritance diagram for Land:



### 3.19.1 \*

Public Member Functions

- virtual [~Land](#) ()=0
- bool [isFacility](#) () const
- void [growGrass](#) ()
- void [removeGrass](#) ()
- bool [isGrassExist](#) () const

### 3.19.2 \*

Additional Inherited Members

### 3.19.3 Detailed Description

[Land](#) merupakan kelas turunan dari [Cell](#) yang merepresentasikan petak-petak yang bisa ditempati oleh [LivingThing](#)

### 3.19.4 Constructor & Destructor Documentation

#### 3.19.4.1 ~Land()

```
virtual Land::~~Land ( ) [pure virtual]
```

Destructor [Land](#)

### 3.19.5 Member Function Documentation

#### 3.19.5.1 growGrass()

```
void Land::growGrass ( ) [virtual]
```

Membuat existGrass menjadi true

Reimplemented from [Cell](#).

### 3.19.5.2 isFacility()

```
bool Land::isFacility ( ) const [virtual]
```

Return true bila [Land](#) adalah sebuah facility

Implements [Cell](#).

### 3.19.5.3 isGrassExist()

```
bool Land::isGrassExist ( ) const [virtual]
```

Mengembalikan keberadaan grass

Implements [Cell](#).

### 3.19.5.4 removeGrass()

```
void Land::removeGrass ( ) [virtual]
```

Membuat existGrass menjadi false

Reimplemented from [Cell](#).

The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Land.h

## 3.20 LinkedList< T > Class Template Reference

```
#include <LinkedList.h>
```

### 3.20.1 \*

Public Member Functions

- [LinkedList](#) ()
- [LinkedList](#) (std::initializer\_list< T > args)
- [LinkedList](#) (const [LinkedList](#)< T > &l)
- [~LinkedList](#) ()
- [LinkedList](#)< T > & [operator=](#) (const [LinkedList](#)< T > &l)
- int [find](#) (T elm)
- int [findPointer](#) (T elm)
- bool [isEmpty](#) () const
- void [add](#) (T elm)
- void [remove](#) (T elm)
- void [removeIdx](#) (int idx)
- T & [get](#) (int idx)
- T & [operator\[\]](#) (int idx)
- void [print](#) ()
- int [len](#) ()

### 3.20.2 Detailed Description

```
template<class T>
class LinkedList< T >
```

Tipe data [LinkedList](#), diimplementasi secara rekursif dengan [LinkedListNode](#)

### 3.20.3 Constructor & Destructor Documentation

#### 3.20.3.1 `LinkedList()` [1/3]

```
template<class T >
LinkedList< T >::LinkedList ( )
```

Konstruktor default [LinkedList](#), membuat empty list

#### 3.20.3.2 `LinkedList()` [2/3]

```
template<class T>
LinkedList< T >::LinkedList (
    std::initializer_list< T > args )
```

Konstruktor dengan initializer list

#### 3.20.3.3 `LinkedList()` [3/3]

```
template<class T>
LinkedList< T >::LinkedList (
    const LinkedList< T > & l )
```

Copy constructor [LinkedList](#)

#### 3.20.3.4 `~LinkedList()`

```
template<class T >
LinkedList< T >::~~LinkedList ( )
```

Destructor [LinkedList](#)

### 3.20.4 Member Function Documentation

#### 3.20.4.1 `add()`

```
template<class T>
void LinkedList< T >::add (
    T elm )
```

Menambah elm sebagai elemen terakhir

#### 3.20.4.2 find()

```
template<class T>
int LinkedList< T >::find (
    T elm )
```

Mencari indeks pertama dari elm dari [LinkedList](#). Jika tidak ada, bernilai -1.

#### 3.20.4.3 findPointer()

```
template<class T>
int LinkedList< T >::findPointer (
    T elm )
```

Mencari indeks pertama dari (\*elm) dari [LinkedList](#) of pointer to Object. Jika tidak ada, bernilai -1.

#### 3.20.4.4 get()

```
template<class T >
T & LinkedList< T >::get (
    int idx )
```

Mengembalikan elemen berindeks idx. Jika diluar range, melempar "Index out of range".

#### 3.20.4.5 isEmpty()

```
template<class T >
bool LinkedList< T >::isEmpty ( ) const
```

Mengembalikan apakah list empty atau tidak

#### 3.20.4.6 len()

```
template<class T >
int LinkedList< T >::len ( )
```

Mengembalikan panjang dari list

#### 3.20.4.7 operator=()

```
template<class T>
LinkedList< T > & LinkedList< T >::operator= (
    const LinkedList< T > & l )
```

Operator= [LinkedList](#)

#### 3.20.4.8 operator[]()

```
template<class T >
T & LinkedList< T >::operator[] (
    int idx )
```

Mengembalikan reference ke elemen berindeks idx. Jika diluar range, melempar "Index out of range".

#### 3.20.4.9 `print()`

```
template<class T >
void LinkedList< T >::print ( )
```

Menampilkan isi dari list ke layar

#### 3.20.4.10 `remove()`

```
template<class T>
void LinkedList< T >::remove (
    T elm )
```

Menghapus keberadaan pertama elm

#### 3.20.4.11 `removeIdx()`

```
template<class T >
void LinkedList< T >::removeIdx (
    int idx )
```

Menghapus elemen berindeks idx. Jika diluar range, melempar "Index out of range".

The documentation for this class was generated from the following file:

- `include/EngiFarm/LinkedList.h`

## 3.21 `LinkedListNode< T >` Class Template Reference

```
#include <LinkedList.h>
```

### 3.21.1 \*

Public Member Functions

- [LinkedListNode](#) (T \_head, [LinkedListNode](#)< T > \*\_next=nullptr)
- [~LinkedListNode](#) ()

### 3.21.2 \*

Public Attributes

- friend [LinkedList](#)< T >

### 3.21.3 Detailed Description

```
template<class T>
class LinkedListNode< T >
```

[LinkedList](#) adalah kelas generik yang merepresentasikan daftar suatu objek Forward declaration dari kelas [LinkedListNode](#)

Anggota kelas implementasi [LinkedList](#) secara rekursifs

### 3.21.4 Constructor & Destructor Documentation

#### 3.21.4.1 LinkedListNode()

```
template<class T>
LinkedListNode< T >::LinkedListNode (
    T _head,
    LinkedListNode< T > * _next = nullptr )
```

Konstruktur `LinkedListNode` dengan argume, deafult tail = nullptr

#### 3.21.4.2 ~LinkedListNode()

```
template<class T >
LinkedListNode< T >::~~LinkedListNode ( )
```

dtor

### 3.21.5 Member Data Documentation

#### 3.21.5.1 LinkedList< T >

```
template<class T>
friend LinkedListNode< T >::LinkedList< T >
```

Membuat `LinkedList` dapat mengakses head dan tail

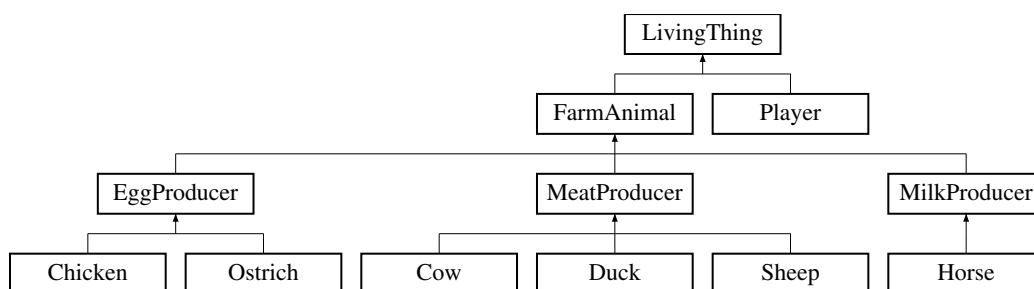
The documentation for this class was generated from the following file:

- include/EngiFarm/LinkedList.h

## 3.22 LivingThing Class Reference

```
#include <LivingThing.h>
```

Inheritance diagram for LivingThing:



#### 3.22.1 \*

Public Member Functions

- `LivingThing` (`Point` position, `Cell ***&worldMap`, `int nRowCell`, `int nColumnCell`)
- `virtual ~LivingThing` ()=0
- `Point getPosition` () const
- `void move` (Direction toWhere)
- `virtual char render` ()=0



### 3.22.2 \*

#### Protected Attributes

- [Cell](#) \*\*\*& [worldMap](#)
- int [nRowCell](#)
- int [nCollumnCell](#)

### 3.22.3 Detailed Description

[LivingThing](#) adalah kelas abstrak yang merepresentasikan makhluk hidup berupa [Player](#) dan [FarmAnimal](#)

### 3.22.4 Constructor & Destructor Documentation

#### 3.22.4.1 LivingThing()

```
LivingThing::LivingThing (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor [LivingThing](#)

#### 3.22.4.2 ~LivingThing()

```
virtual LivingThing::~LivingThing ( ) [pure virtual]
```

Destructor dari [LivingThing](#)

### 3.22.5 Member Function Documentation

#### 3.22.5.1 getPosition()

```
Point LivingThing::getPosition ( ) const
```

Mengembalikan position

#### 3.22.5.2 move()

```
void LivingThing::move (
    Direction toWhere )
```

Berpindah ke suatu lokasi. Apabila tidak bisa (!canMoveTo), throw "Cannot move to the direction".

#### 3.22.5.3 render()

```
virtual char LivingThing::render ( ) [pure virtual]
```

Mengembalikan char untuk dirender ke layar

Implemented in [Player](#), [Chicken](#), [Cow](#), [Duck](#), [Horse](#), [Ostrich](#), and [Sheep](#).

### 3.22.6 Member Data Documentation

#### 3.22.6.1 nColumnCell

```
int LivingThing::nColumnCell [protected]
```

Nilai efektif kolom untuk Matriks [Cell](#)

#### 3.22.6.2 nRowCell

```
int LivingThing::nRowCell [protected]
```

Nilai efektif baris untuk Matriks [Cell](#)

#### 3.22.6.3 worldMap

```
Cell***& LivingThing::worldMap [protected]
```

Representasi dunia tempat [LivingThing](#) tinggal

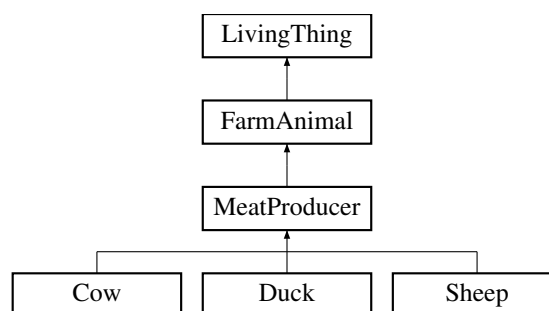
The documentation for this class was generated from the following file:

- include/EngiFarm/LivingThing.h

## 3.23 MeatProducer Class Reference

```
#include <MeatProducer.h>
```

Inheritance diagram for MeatProducer:



### 3.23.1 \*

Public Member Functions

- [MeatProducer](#) (int \_maxTimeToGetHungry, [Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nColumnCell)
- virtual [~MeatProducer](#) ()=0
- bool [getKillable](#) ()
- bool [getProduce](#) ()

### 3.23.2 \*

Additional Inherited Members

### 3.23.3 Detailed Description

[MeatProducer](#) adalah kelas abstrak turunan dari [FarmAnimal](#) yang tinggal di barn dan dapat menghasilkan daging jika dilakukan aksi kill

### 3.23.4 Constructor & Destructor Documentation

#### 3.23.4.1 MeatProducer()

```
MeatProducer::MeatProducer (
    int _maxTimeToGetHungry,
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H

#### 3.23.4.2 ~MeatProducer()

```
virtual MeatProducer::~~MeatProducer ( ) [pure virtual]
```

Penerusan overloading (virtual) destruktur

### 3.23.5 Member Function Documentation

#### 3.23.5.1 getKillable()

```
bool MeatProducer::getKillable ( ) [virtual]
```

Mengembalikan nilai dari killable

Implements [FarmAnimal](#).

#### 3.23.5.2 getProduce()

```
bool MeatProducer::getProduce ( ) [virtual]
```

Mengembalikan false karena [MeatProducer](#) tidak bisa di Interact

Implements [FarmAnimal](#).

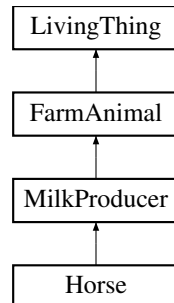
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/MeatProducer.h

## 3.24 MilkProducer Class Reference

```
#include <MilkProducer.h>
```

Inheritance diagram for MilkProducer:



### 3.24.1 \*

Public Member Functions

- [MilkProducer](#) (int \_maxTimeToGetHungry, [Point](#) position, [Cell](#) \*\*\*&[worldMap](#), int nRowCell, int nColumnCell)
- virtual [~MilkProducer](#) ()=0
- bool [getProduce](#) ()
- bool [getKillable](#) ()

### 3.24.2 \*

Protected Attributes

- bool [canProduce](#) = false

### 3.24.3 \*

Additional Inherited Members

## 3.24.4 Detailed Description

[MilkProducer](#) adalah kelas abstrak turunan dari [FarmAnimal](#) yang tinggal di [GrassLand](#) dan dapat menghasilkan [Product](#) berupa Milk jika di Interact

## 3.24.5 Constructor & Destructor Documentation

### 3.24.5.1 MilkProducer()

```
MilkProducer::MilkProducer (
    int _maxTimeToGetHungry,
```

```
Point position,  
Cell ***& worldMap,  
int nRowCell,  
int nCollumnCell )
```

Constructor maxTimeToGetHungry dengan nilai H

#### 3.24.5.2 ~MilkProducer()

```
virtual MilkProducer::~MilkProducer ( ) [pure virtual]
```

Penerusan overloading (virtual) destruktur

### 3.24.6 Member Function Documentation

#### 3.24.6.1 getKillable()

```
bool MilkProducer::getKillable ( ) [virtual]
```

Mengembalikan false karena [MilkProducer](#) tidak bisa di kill

Implements [FarmAnimal](#).

#### 3.24.6.2 getProduce()

```
bool MilkProducer::getProduce ( ) [virtual]
```

Mengembalikan nilai dari canProduce

Implements [FarmAnimal](#).

### 3.24.7 Member Data Documentation

#### 3.24.7.1 canProduce

```
bool MilkProducer::canProduce = false [protected]
```

Menentukan apakah [FarmAnimal](#) dapat menghasilkan produk apabila diinteract

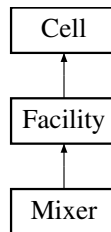
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/MilkProducer.h

## 3.25 Mixer Class Reference

```
#include <Mixer.h>
```

Inheritance diagram for Mixer:



### 3.25.1 \*

Public Member Functions

- [Category getCategory](#) () const

### 3.25.2 \*

Additional Inherited Members

### 3.25.3 Detailed Description

[Mixer](#) merupakan kelas turunan dari [Facility](#) yang digunakan untuk membuat [SideProduct](#)

### 3.25.4 Member Function Documentation

#### 3.25.4.1 getCategory()

```
Category Mixer::getCategory ( ) const [virtual]
```

Return kategori dari objek ini

Implements [Cell](#).

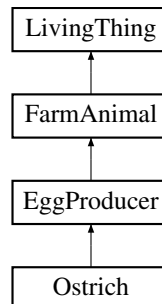
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Mixer.h

## 3.26 Ostrich Class Reference

```
#include <Ostrich.h>
```

Inheritance diagram for Ostrich:



### 3.26.1 \*

Public Member Functions

- [Ostrich](#) ([Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nCollumnCell)
- [FarmProduct](#) \* [produceProduct](#) ([Action](#))
- std::string [makeNoise](#) () const
- char [render](#) ()

### 3.26.2 \*

Additional Inherited Members

### 3.26.3 Detailed Description

[Ostrich](#) merupakan kelas turunan dari [EggProducer](#) yang menghasilkan [OstrichEgg](#) saat diinteract

### 3.26.4 Constructor & Destructor Documentation

#### 3.26.4.1 Ostrich()

```
Ostrich::Ostrich (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor

### 3.26.5 Member Function Documentation

### 3.26.5.1 makeNoise()

```
std::string Ostrich::makeNoise ( ) const [virtual]
```

Mengembalikan suara dari [Chicken](#)

Implements [FarmAnimal](#).

### 3.26.5.2 produceProduct()

```
FarmProduct* Ostrich::produceProduct (
    Action ) [virtual]
```

Mengembalikan FarmProduk yang akan dihasilkan [Ostrich](#) bila [Ostrich](#) di interact

Implements [FarmAnimal](#).

### 3.26.5.3 render()

```
char Ostrich::render ( ) [virtual]
```

Mengembalikan karakter yang merepresentasikan [Ostrich](#) saat Hungry dan tidak Hungry

Implements [LivingThing](#).

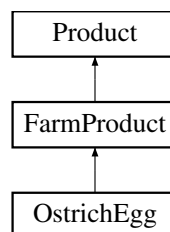
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/Ostrich.h

## 3.27 OstrichEgg Class Reference

```
#include <OstrichEgg.h>
```

Inheritance diagram for OstrichEgg:



### 3.27.1 \*

Public Member Functions

- int [getPrice](#) ( ) const
- [Category getCategory](#) ( ) const



## 3.27.2 \*

Additional Inherited Members

## 3.27.3 Detailed Description

[OstrichEgg](#) adalah kelas turunan dari [FarmProduct](#) yang dihasilkan dengan interact dengan [Ostrich](#)

## 3.27.4 Member Function Documentation

## 3.27.4.1 getCategory()

```
Category OstrichEgg::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

## 3.27.4.2 getPrice()

```
int OstrichEgg::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

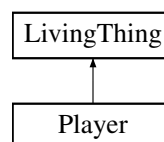
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/OstrichEgg.h

## 3.28 Player Class Reference

```
#include <Player.h>
```

Inheritance diagram for Player:



## 3.28.1 \*

Public Member Functions

- [Player](#) ([Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nColumnCell)
- [~Player](#) ()
- void [talk](#) ([LinkedList](#)< [FarmAnimal](#) \* > &farmAnimal, [LinkedList](#)< std::string > &mesQueue)
- void [interact](#) ([LinkedList](#)< [FarmAnimal](#) \* > &farmAnimal)
- void [kill](#) ([LinkedList](#)< [FarmAnimal](#) \* > &farmAnimal, int &nAnimal)
- void [grow](#) ([LinkedList](#)< std::string > &mesQueue)
- void [mix](#) ([LinkedList](#)< std::string > &mesQueue)
- char [render](#) ()
- void [takeWater](#) ()
- void [sellAll](#) ()
- int [getMoney](#) ()
- int [getWater](#) ()
- [LinkedList](#)< [Product](#) \* > & [getInventory](#) ()
- [LinkedList](#)< [SideProduct](#) \* > [getrecipeBook](#) ()

### 3.28.2 \*

Additional Inherited Members

### 3.28.3 Detailed Description

[Player](#) adalah kelas yang merepresentasikan pemain dengan semua aksinya di dunia

### 3.28.4 Constructor & Destructor Documentation

#### 3.28.4.1 [Player\(\)](#)

```
Player::Player (
    Point position,
    Cell ***& worldMap,
    int nRowCell,
    int nCollumnCell )
```

Constructor [Player](#) di position, recipeBook diinisialisasi dengan semua [SideProduct](#) yang terdefinisi

#### 3.28.4.2 [~Player\(\)](#)

```
Player::~~Player ( )
```

Destructor [Player](#)

### 3.28.5 Member Function Documentation

#### 3.28.5.1 [getInventory\(\)](#)

```
LinkedList<Product*>& Player::getInventory ( )
```

Getter inventory yang dipegang [Player](#)

#### 3.28.5.2 [getMoney\(\)](#)

```
int Player::getMoney ( )
```

Getter banyak uang yang dimiliki [Player](#)

#### 3.28.5.3 [getrecipeBook\(\)](#)

```
LinkedList<SideProduct*> Player::getrecipeBook ( )
```

Getter daftar resep yang dimiliki [Player](#)

#### 3.28.5.4 [getWater\(\)](#)

```
int Player::getWater ( )
```

Getter banyak air yang dimiliki [Player](#)

#### 3.28.5.5 grow()

```
void Player::grow (
    LinkedList< std::string > & mesQueue )
```

Menumbuhkan rumput pada cell yang sedang ditempati oleh [Player](#)

#### 3.28.5.6 interact()

```
void Player::interact (
    LinkedList< FarmAnimal * > & farmAnimal )
```

[Player](#) mengambil [FarmProduct](#) dari semua [FarmAnimal](#) terdekat tanpa membunuh [FarmAnimal](#) tersebut. Bekerja untuk [FarmAnimal](#) jenis MilkProducing dan EggProducing. Contoh [FarmProduct](#) : [ChickenEgg](#), [CowMilk](#).

#### 3.28.5.7 kill()

```
void Player::kill (
    LinkedList< FarmAnimal * > & farmAnimal,
    int & nAnimal )
```

[Player](#) mengambil [FarmProduct](#) dari semua [FarmAnimal](#) terdekat dengan cara membunuh [FarmAnimal](#) tersebut. Bekerja untuk [FarmAnimal](#) jenis MeatProducing. Contoh [FarmProduct](#) : [CowMeat](#), [ChickenMeat](#).

#### 3.28.5.8 mix()

```
void Player::mix (
    LinkedList< std::string > & mesQueue )
```

Menciptakan [SideProduct](#) dari [FarmProduct](#) bila [Player](#) dekat dengan mixer

#### 3.28.5.9 render()

```
char Player::render ( ) [virtual]
```

Mengembalikan char untuk dirender ke layar

Implements [LivingThing](#).

#### 3.28.5.10 sellAll()

```
void Player::sellAll ( )
```

Menjual semua product di inventory

#### 3.28.5.11 takeWater()

```
void Player::takeWater ( )
```

Mengambil air dari well

#### 3.28.5.12 talk()

```
void Player::talk (
    LinkedList< FarmAnimal * > & farmAnimal,
    LinkedList< std::string > & mesQueue )
```

[Player](#) berbicara dengan semua [FarmAnimal](#) terdekat.

The documentation for this class was generated from the following file:

- include/EngiFarm/Player.h

## 3.29 Point Struct Reference

```
#include <Point.h>
```

### 3.29.1 \*

#### Public Attributes

- int [x](#)
- int [y](#)

### 3.29.2 Detailed Description

[Point](#) adalah suatu struktur data yang menyimpan posisi di bidang 2 dimensi

### 3.29.3 Member Data Documentation

#### 3.29.3.1 x

```
int Point::x
```

Absis dari poin

#### 3.29.3.2 y

```
int Point::y
```

Ordinat dari poin

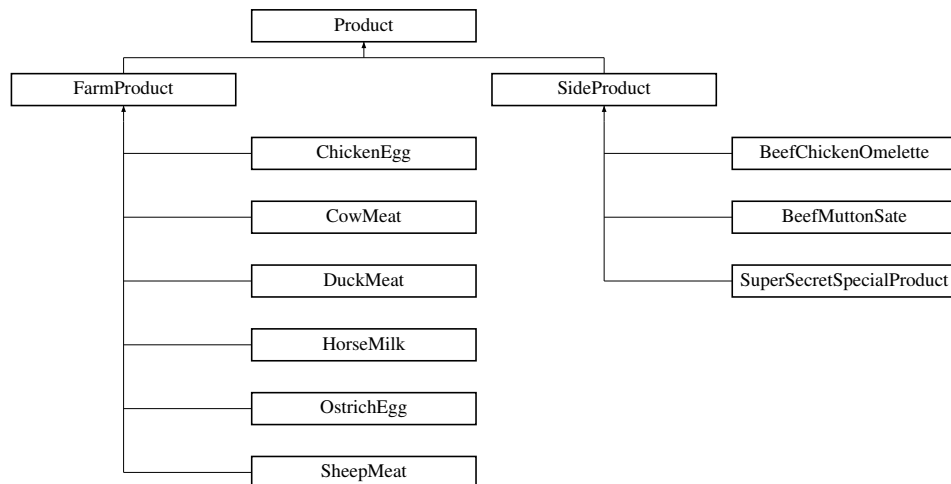
The documentation for this struct was generated from the following file:

- `include/EngiFarm/Point.h`

## 3.30 Product Class Reference

```
#include <Product.h>
```

Inheritance diagram for Product:



### 3.30.1 \*

Public Types

- enum [Category](#) {  
**CHICKENEGG, BEEFCHICKENOMELETTE, BEEFMUTONSATE, COWMEAT,  
 DUCKMEAT, HORSEMILK, OSTRICHEGG, SHEEPMEAT,  
 SUPERSECRETSPECIALPRODUCT** }

### 3.30.2 \*

Public Member Functions

- virtual int [getPrice](#) () const =0
- virtual [Category](#) [getCategory](#) () const =0
- bool [operator==](#) ([Product](#) &P)
- bool [operator!=](#) ([Product](#) &P)

### 3.30.3 Detailed Description

[Product](#) adalah kelas abstrak yang merepresentasikan produk yang bisa dibuat dan dijual player

### 3.30.4 Member Enumeration Documentation

### 3.30.4.1 Category

```
enum Product::Category
```

enumerasi kategori dari suatu produk, return value dari getCategory

## 3.30.5 Member Function Documentation

### 3.30.5.1 getCategory()

```
virtual Category Product::getCategory ( ) const [pure virtual]
```

mengembalikan kategori dari produk ini

Implemented in [BeefChickenOmelette](#), [BeefMuttonSate](#), [SuperSecretSpecialProduct](#), [ChickenEgg](#), [CowMeat](#), [HorseMilk](#), [OstrichEgg](#), [SheepMeat](#), and [DuckMeat](#).

### 3.30.5.2 getPrice()

```
virtual int Product::getPrice ( ) const [pure virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implemented in [BeefChickenOmelette](#), [BeefMuttonSate](#), [SuperSecretSpecialProduct](#), [ChickenEgg](#), [CowMeat](#), [HorseMilk](#), [OstrichEgg](#), [SheepMeat](#), and [DuckMeat](#).

### 3.30.5.3 operator"!="()

```
bool Product::operator!= (
    Product & P )
```

Mengembalikan hasil perbandingan dereference

### 3.30.5.4 operator=="()

```
bool Product::operator== (
    Product & P )
```

Mengembalikan hasil perbandingan dereference

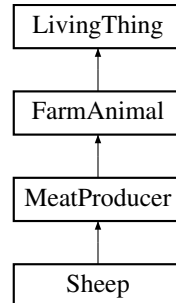
The documentation for this class was generated from the following file:

- [include/EngiFarm/Product/Product.h](#)

## 3.31 Sheep Class Reference

```
#include <Sheep.h>
```

Inheritance diagram for Sheep:



### 3.31.1 \*

Public Member Functions

- [Sheep](#) ([Point](#) position, [Cell](#) \*\*\*&worldMap, int nRowCell, int nCollumnCell)
- [FarmProduct](#) \* [produceProduct](#) ([Action](#))
- [std::string](#) [makeNoise](#) () const
- [char](#) [render](#) ()

### 3.31.2 \*

Additional Inherited Members

### 3.31.3 Detailed Description

[Sheep](#) merupakan kelas turunan dari [MeatProducer](#) yang menghasilkan [SheepMeat](#) saat diinteract

### 3.31.4 Constructor & Destructor Documentation

#### 3.31.4.1 Sheep()

```
Sheep::Sheep (  
    Point position,  
    Cell ***& worldMap,  
    int nRowCell,  
    int nCollumnCell )
```

Constructor

### 3.31.5 Member Function Documentation

### 3.31.5.1 makeNoise()

```
std::string Sheep::makeNoise ( ) const [virtual]
```

Mengembalikan suara dari [Sheep](#)

Implements [FarmAnimal](#).

### 3.31.5.2 produceProduct()

```
FarmProduct* Sheep::produceProduct (
    Action ) [virtual]
```

Mengembalikan FarmProduk yang akan dihasilkan [Sheep](#) bila [Sheep](#) di kill

Implements [FarmAnimal](#).

### 3.31.5.3 render()

```
char Sheep::render ( ) [virtual]
```

Mengembalikan karakter yang merepresentasikan [Sheep](#) saat Hungry dan tidak Hungry

Implements [LivingThing](#).

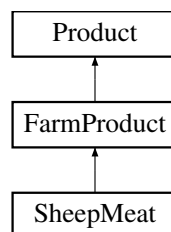
The documentation for this class was generated from the following file:

- include/EngiFarm/FarmAnimal/Sheep.h

## 3.32 SheepMeat Class Reference

```
#include <SheepMeat.h>
```

Inheritance diagram for SheepMeat:



### 3.32.1 \*

Public Member Functions

- int [getPrice](#) ( ) const
- [Category getCategory](#) ( ) const



## 3.32.2 \*

Additional Inherited Members

## 3.32.3 Detailed Description

[SheepMeat](#) adalah kelas turunan dari [FarmProduct](#) yang dihasilkan dengan kill [Sheep](#)

## 3.32.4 Member Function Documentation

## 3.32.4.1 getCategory()

```
Category SheepMeat::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

## 3.32.4.2 getPrice()

```
int SheepMeat::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

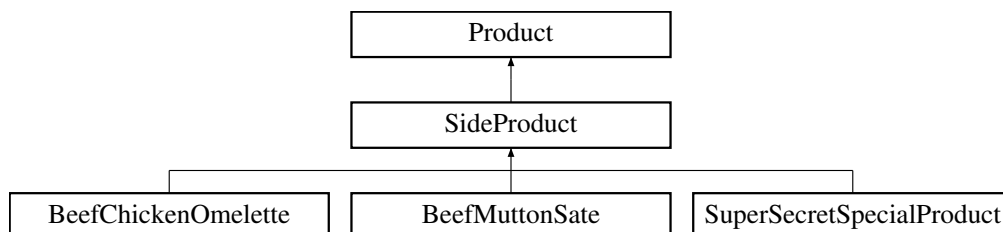
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/SheepMeat.h

## 3.33 SideProduct Class Reference

```
#include <SideProduct.h>
```

Inheritance diagram for SideProduct:



## 3.33.1 \*

Public Member Functions

- virtual [LinkedList< Product \\* >](#) & [getRecipe](#) ()=0
- virtual [~SideProduct](#) ()

### 3.33.2 \*

Additional Inherited Members

### 3.33.3 Detailed Description

[SideProduct](#) adalah kelas abstrak turunan dari kelas [Product](#) yang didapat dari hasil mix

### 3.33.4 Constructor & Destructor Documentation

#### 3.33.4.1 ~SideProduct()

```
virtual SideProduct::~~SideProduct ( ) [virtual]
```

Destructor side product

### 3.33.5 Member Function Documentation

#### 3.33.5.1 getRecipe()

```
virtual LinkedList<Product*>& SideProduct::getRecipe ( ) [pure virtual]
```

Mengembalikan resep dari produk

Implemented in [BeefChickenOmelette](#), [BeefMuttonSate](#), and [SuperSecretSpecialProduct](#).

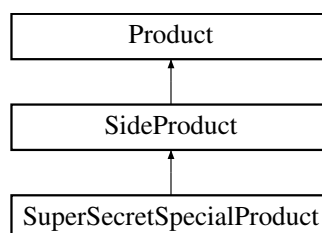
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/SideProduct.h

## 3.34 SuperSecretSpecialProduct Class Reference

```
#include <SuperSecretSpecialProduct.h>
```

Inheritance diagram for SuperSecretSpecialProduct:



### 3.34.1 \*

Public Member Functions

- [SuperSecretSpecialProduct](#) ( )
- [int getPrice](#) ( ) const
- [Category getCategory](#) ( ) const
- [LinkedList< Product \\* > & getRecipe](#) ( )

### 3.34.2 \*

Additional Inherited Members

### 3.34.3 Detailed Description

[SuperSecretSpecialProduct](#) adalah kelas turunan dari [SideProduct](#) yang dihasilkan dengan mix [HorseMilk](#) dan [OstrichEgg](#)

### 3.34.4 Constructor & Destructor Documentation

#### 3.34.4.1 SuperSecretSpecialProduct()

```
SuperSecretSpecialProduct::SuperSecretSpecialProduct ( )
```

Constructor untuk inisialisasi recipe

### 3.34.5 Member Function Documentation

#### 3.34.5.1 getCategory()

```
Category SuperSecretSpecialProduct::getCategory ( ) const [virtual]
```

Mengembalikan category dari produk

Implements [Product](#).

#### 3.34.5.2 getPrice()

```
int SuperSecretSpecialProduct::getPrice ( ) const [virtual]
```

getPrice mengembalikan harga yang didefinisikan

Implements [Product](#).

#### 3.34.5.3 getRecipe()

```
LinkedList<Product*>& SuperSecretSpecialProduct::getRecipe ( ) [virtual]
```

Mengembalikan resep dari produk

Implements [SideProduct](#).

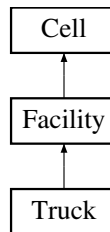
The documentation for this class was generated from the following file:

- include/EngiFarm/Product/SuperSecretSpecialProduct.h

### 3.35 Truck Class Reference

```
#include <Truck.h>
```

Inheritance diagram for Truck:



#### 3.35.1 \*

Public Member Functions

- [Category getCategory \(\)](#) const

#### 3.35.2 \*

Additional Inherited Members

### 3.35.3 Detailed Description

[Truck](#) merupakan kelas turunan dari [Facility](#) yang dapat digunakan untuk menjual seluruh barang di inventori

### 3.35.4 Member Function Documentation

#### 3.35.4.1 getCategory()

```
Category Truck::getCategory ( ) const [virtual]
```

Return kategori dari objek ini

Implements [Cell](#).

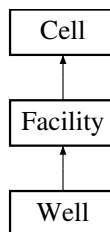
The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Truck.h

## 3.36 Well Class Reference

```
#include <Well.h>
```

Inheritance diagram for Well:



### 3.36.1 \*

Public Member Functions

- [Category getCategory](#) () const

### 3.36.2 \*

Additional Inherited Members

### 3.36.3 Detailed Description

[Well](#) merupakan kelas turunan dari [Facility](#) yang digunakan untuk memberi Water untuk [Player](#)

### 3.36.4 Member Function Documentation

#### 3.36.4.1 getCategory()

```
Category Well::getCategory ( ) const [virtual]
```

Return kategori dari objek ini

Implements [Cell](#).

The documentation for this class was generated from the following file:

- include/EngiFarm/Cell/Well.h

### 3.37 World Class Reference

```
#include <World.h>
```

#### 3.37.1 \*

##### Public Member Functions

- [World](#) ()
- [~World](#) ()
- void [Input](#) ()
- void [Update](#) ()
- void [Draw](#) ()

#### 3.37.2 Detailed Description

[World](#) adalah kelas yang merepresentasikan dunia yang menyimpan semua [Cell](#) dan [LivingThing](#) di dalamnya

#### 3.37.3 Constructor & Destructor Documentation

##### 3.37.3.1 [World](#)()

```
World::World ( )
```

Constructor [World](#). Memanggil ctor dan menginisialisasi semua atribut world; Pertama, map diinisialisasi sesuai dengan spesifikasi, saat penginisialisasian map, ctor untuk object riil dari cell seperti coop, barn, dan well dipanggil Kedua, ctor [Player](#) dipanggil dengan argumen [Point](#) lokasi awal player dan reference ke map yang sudah didefinisikan pada tahap pertama Terakhir, animalList diinisialisasi dengan beberapa [FarmAnimal](#) secara random

##### 3.37.3.2 [~World](#)()

```
World::~~World ( )
```

Destructor [World](#). Dealokasi seluruh [Cell](#) dan [FarmAnimal](#), termasuk seluruh pointer yang berhubungan.

#### 3.37.4 Member Function Documentation

##### 3.37.4.1 [Draw](#)()

```
void World::Draw ( )
```

Megambarkan representasi state program ([World](#)) seperti lokasi setiap objek, money, water, dan Inventory [Player](#), dsb ke layar.

##### 3.37.4.2 [Input](#)()

```
void World::Input ( )
```

Membaca input user dari stdin lalu melakukan aksi sesuai dengan spesifikasi, misal, input == MOVELEFT, maka akan dipanggil pl.move(LEFT). Bila input == INTERACT, maka akan dipanggil pl.interact(animalList), dsb.

##### 3.37.4.3 [Update](#)()

```
void World::Update ( )
```

Pada [World::Update\(\)](#), setiap fungsi yang dipanggil secara berkala seperti [FarmAnimal::tick\(\)](#) akan dipanggil.

The documentation for this class was generated from the following file:

- include/EngiFarm/World.h

# Index

- ~BeefChickenOmelette
  - BeefChickenOmelette, [6](#)
- ~BeefMuttonSate
  - BeefMuttonSate, [8](#)
- ~Cell
  - Cell, [10](#)
- ~EggProducer
  - EggProducer, [21](#)
- ~Facility
  - Facility, [23](#)
- ~FarmAnimal
  - FarmAnimal, [25](#)
- ~Land
  - Land, [31](#)
- ~LinkedList
  - LinkedList< T >, [33](#)
- ~LinkedListNode
  - LinkedListNode< T >, [36](#)
- ~LivingThing
  - LivingThing, [37](#)
- ~MeatProducer
  - MeatProducer, [39](#)
- ~MilkProducer
  - MilkProducer, [41](#)
- ~Player
  - Player, [46](#)
- ~SideProduct
  - SideProduct, [54](#)
- ~World
  - World, [58](#)
- Action
  - FarmAnimal, [25](#)
- add
  - LinkedList< T >, [33](#)
- Barn, [5](#)
  - getCategory, [5](#)
- BeefChickenOmelette, [6](#)
  - ~BeefChickenOmelette, [6](#)
  - BeefChickenOmelette, [6](#)
  - getCategory, [7](#)
  - getPrice, [7](#)
  - getRecipe, [7](#)
- BeefMuttonSate, [7](#)
  - ~BeefMuttonSate, [8](#)
  - BeefMuttonSate, [8](#)
  - getCategory, [8](#)
  - getPrice, [8](#)
  - getRecipe, [8](#)

- canProduce
  - EggProducer, [22](#)
  - MilkProducer, [41](#)
- Category
  - Cell, [10](#)
  - Product, [49](#)
- Cell, [9](#)
  - ~Cell, [10](#)
  - Category, [10](#)
  - getCategory, [10](#)
  - getIsOccupied, [10](#)
  - growGrass, [10](#)
  - isFacility, [10](#)
  - isGrassExist, [11](#)
  - isOccupied, [11](#)
  - removeGrass, [11](#)
  - setIsOccupied, [11](#)
- Chicken, [12](#)
  - Chicken, [12](#)
  - makeNoise, [12](#)
  - produceProduct, [13](#)
  - render, [13](#)
- ChickenEgg, [13](#)
  - getCategory, [14](#)
  - getPrice, [14](#)
- Coop, [14](#)
  - getCategory, [15](#)
- Cow, [15](#)
  - Cow, [16](#)
  - makeNoise, [16](#)
  - produceProduct, [16](#)
  - render, [16](#)
- CowMeat, [17](#)
  - getCategory, [17](#)
  - getPrice, [17](#)
- Draw
  - World, [58](#)
- Duck, [18](#)
  - Duck, [18](#)
  - makeNoise, [18](#)
  - produceProduct, [19](#)
  - render, [19](#)
- DuckMeat, [19](#)
  - getCategory, [20](#)
  - getPrice, [20](#)
- eat
  - FarmAnimal, [25](#)
- EggProducer, [20](#)

- ~EggProducer, 21
- canProduce, 22
- EggProducer, 21
- getKillable, 21
- getProduce, 21
- Facility, 22
  - ~Facility, 23
  - Facility, 23
  - isFacility, 23
  - isGrassExist, 23
- FarmAnimal, 24
  - ~FarmAnimal, 25
  - Action, 25
  - eat, 25
  - FarmAnimal, 25
  - getKillable, 25
  - getProduce, 26
  - isDead, 26
  - isHungry, 26
  - makeNoise, 26
  - maxTimeToGetHungry, 26
  - produceProduct, 26
  - tick, 26
  - timeToGetHungry, 26
- FarmProduct, 27
- find
  - LinkedList< T >, 34
- findPointer
  - LinkedList< T >, 34
- get
  - LinkedList< T >, 34
- getCategory
  - Barn, 5
  - BeefChickenOmelette, 7
  - BeefMuttonSate, 8
  - Cell, 10
  - ChickenEgg, 14
  - Coop, 15
  - CowMeat, 17
  - DuckMeat, 20
  - GrassLand, 28
  - HorseMilk, 30
  - Mixer, 42
  - OstrichEgg, 45
  - Product, 50
  - SheepMeat, 53
  - SuperSecretSpecialProduct, 55
  - Truck, 56
  - Well, 57
- getInventory
  - Player, 46
- getIsOccupied
  - Cell, 10
- getKillable
  - EggProducer, 21
  - FarmAnimal, 25
  - MeatProducer, 39
  - MilkProducer, 41
- getMoney
  - Player, 46
- getPosition
  - LivingThing, 37
- getPrice
  - BeefChickenOmelette, 7
  - BeefMuttonSate, 8
  - ChickenEgg, 14
  - CowMeat, 17
  - DuckMeat, 20
  - HorseMilk, 30
  - OstrichEgg, 45
  - Product, 50
  - SheepMeat, 53
  - SuperSecretSpecialProduct, 55
- getProduce
  - EggProducer, 21
  - FarmAnimal, 26
  - MeatProducer, 39
  - MilkProducer, 41
- getRecipe
  - BeefChickenOmelette, 7
  - BeefMuttonSate, 8
  - SideProduct, 54
  - SuperSecretSpecialProduct, 55
- getrecipeBook
  - Player, 46
- getWater
  - Player, 46
- GrassLand, 27
  - getCategory, 28
- grow
  - Player, 47
- growGrass
  - Cell, 10
  - Land, 31
- Horse, 28
  - Horse, 29
  - makeNoise, 29
  - produceProduct, 29
  - render, 29
- HorseMilk, 30
  - getCategory, 30
  - getPrice, 30
- Input
  - World, 58
- interact
  - Player, 47
- isDead
  - FarmAnimal, 26
- isEmpty
  - LinkedList< T >, 34
- isFacility
  - Cell, 10
  - Facility, 23
  - Land, 32



- isGrassExist
  - Cell, [11](#)
  - Facility, [23](#)
  - Land, [32](#)
- isHungry
  - FarmAnimal, [26](#)
- isOccupied
  - Cell, [11](#)
- kill
  - Player, [47](#)
- Land, [31](#)
  - ~Land, [31](#)
  - growGrass, [31](#)
  - isFacility, [32](#)
  - isGrassExist, [32](#)
  - removeGrass, [32](#)
- len
  - LinkedList< T >, [34](#)
- LinkedList
  - LinkedList< T >, [33](#)
- LinkedList< T >, [32](#)
  - ~LinkedList, [33](#)
  - add, [33](#)
  - find, [34](#)
  - findPointer, [34](#)
  - get, [34](#)
  - isEmpty, [34](#)
  - len, [34](#)
  - LinkedList, [33](#)
  - LinkedListNode< T >, [36](#)
  - operator=, [34](#)
  - operator[], [34](#)
  - print, [34](#)
  - remove, [35](#)
  - removeldx, [35](#)
- LinkedListNode
  - LinkedListNode< T >, [36](#)
- LinkedListNode< T >, [35](#)
  - ~LinkedListNode, [36](#)
  - LinkedList< T >, [36](#)
  - LinkedListNode, [36](#)
- LivingThing, [36](#)
  - ~LivingThing, [37](#)
  - getPosition, [37](#)
  - LivingThing, [37](#)
  - move, [37](#)
  - nColumnCell, [38](#)
  - nRowCell, [38](#)
  - render, [37](#)
  - worldMap, [38](#)
- makeNoise
  - Chicken, [12](#)
  - Cow, [16](#)
  - Duck, [18](#)
  - FarmAnimal, [26](#)
  - Horse, [29](#)
  - Ostrich, [43](#)
  - Sheep, [51](#)
- maxTimeToGetHungry
  - FarmAnimal, [26](#)
- MeatProducer, [38](#)
  - ~MeatProducer, [39](#)
  - getKillable, [39](#)
  - getProduce, [39](#)
  - MeatProducer, [39](#)
- MilkProducer, [40](#)
  - ~MilkProducer, [41](#)
  - canProduce, [41](#)
  - getKillable, [41](#)
  - getProduce, [41](#)
  - MilkProducer, [40](#)
- mix
  - Player, [47](#)
- Mixer, [42](#)
  - getCategory, [42](#)
- move
  - LivingThing, [37](#)
- nColumnCell
  - LivingThing, [38](#)
- nRowCell
  - LivingThing, [38](#)
- operator!=
  - Product, [50](#)
- operator=
  - LinkedList< T >, [34](#)
- operator==
  - Product, [50](#)
- operator[]
  - LinkedList< T >, [34](#)
- Ostrich, [43](#)
  - makeNoise, [43](#)
  - Ostrich, [43](#)
  - produceProduct, [44](#)
  - render, [44](#)
- OstrichEgg, [44](#)
  - getCategory, [45](#)
  - getPrice, [45](#)
- Player, [45](#)
  - ~Player, [46](#)
  - getInventory, [46](#)
  - getMoney, [46](#)
  - getrecipeBook, [46](#)
  - getWater, [46](#)
  - grow, [47](#)
  - interact, [47](#)
  - kill, [47](#)
  - mix, [47](#)
  - Player, [46](#)
  - render, [47](#)
  - sellAll, [47](#)
  - takeWater, [47](#)
  - talk, [47](#)

- Point, [48](#)
  - x, [48](#)
  - y, [48](#)
- print
  - LinkedList< T >, [34](#)
- produceProduct
  - Chicken, [13](#)
  - Cow, [16](#)
  - Duck, [19](#)
  - FarmAnimal, [26](#)
  - Horse, [29](#)
  - Ostrich, [44](#)
  - Sheep, [52](#)
- Product, [49](#)
  - Category, [49](#)
  - getCategory, [50](#)
  - getPrice, [50](#)
  - operator!=, [50](#)
  - operator==, [50](#)
- remove
  - LinkedList< T >, [35](#)
- removeGrass
  - Cell, [11](#)
  - Land, [32](#)
- removeldx
  - LinkedList< T >, [35](#)
- render
  - Chicken, [13](#)
  - Cow, [16](#)
  - Duck, [19](#)
  - Horse, [29](#)
  - LivingThing, [37](#)
  - Ostrich, [44](#)
  - Player, [47](#)
  - Sheep, [52](#)
- sellAll
  - Player, [47](#)
- setIsOccupied
  - Cell, [11](#)
- Sheep, [51](#)
  - makeNoise, [51](#)
  - produceProduct, [52](#)
  - render, [52](#)
  - Sheep, [51](#)
- SheepMeat, [52](#)
  - getCategory, [53](#)
  - getPrice, [53](#)
- SideProduct, [53](#)
  - ~SideProduct, [54](#)
  - getRecipe, [54](#)
- SuperSecretSpecialProduct, [54](#)
  - getCategory, [55](#)
  - getPrice, [55](#)
  - getRecipe, [55](#)
  - SuperSecretSpecialProduct, [55](#)
- takeWater
  - Player, [47](#)
- talk
  - Player, [47](#)
- tick
  - FarmAnimal, [26](#)
- timeToGetHungry
  - FarmAnimal, [26](#)
- Truck, [56](#)
  - getCategory, [56](#)
- Update
  - World, [58](#)
- Well, [57](#)
  - getCategory, [57](#)
- World, [58](#)
  - ~World, [58](#)
  - Draw, [58](#)
  - Input, [58](#)
  - Update, [58](#)
  - World, [58](#)
- worldMap
  - LivingThing, [38](#)
- x
  - Point, [48](#)
- y
  - Point, [48](#)